

Introduction

Name: Sparsh Paliwal

Email: sparshpaliwal123@gmail.com

IRC nick: PaliwalSparsh

Github: <https://github.com/PaliwalSparsh>

Computer Science, as a subject, is something that has intrigued me right from my school days. I chose Computer science as an elective in high school. Since I have always enjoyed acquiring information, learning new concepts and ideas that the courses introduced me to, and stood amongst the best performers in the subject. I decided to pursue my undergraduate degree in the field of Information Technology at National Institute of Technology, Durgapur which is among the most reputed Government Institutions in India. Over the years, Computer science department at NIT, Durgapur has augmented its reputation as a front runner in computer science research and education in India through collaborations with various organisation like CERN and CISCO.

Since childhood I have pursued art for recreation. This meant taking part in various competitions and workshops during school and college.

I am also an active member of Dramatics club at NIT, Durgapur. We won two national level dramatics competitions for street plays (which are meant to spread awareness among individuals by performing plays on streets). I also won competition in mono-acting at Indian Institute of Management, Calcutta. A little while back, I wrote and directed a street play that has received lot of accolades in various competitions. My whole experience at Dramatics club has educated me a lot about teamwork and leadership. Apart from this I have also taught myself to play flute and guitar. Recently I realised my love towards digital art while working on different art projects like designing posters for events ,short films and assets for mobile games. This has encouraged me to take formal education in arts. After completing bachelors degree, I will works towards getting a Masters in Graphic Design or Animation. My ultimate goal is to leverage my knowledge of both arts and technology to help innovate ways to integrate these diverse areas.

I am sure my experience in fields of both art and technology will help me in tackling varying challenges during the course of this project. As my past work covers all areas that are mandatory for anyone willing to work on this project. I am a good programmer, who can design assets, compose music and write beautiful stories.

Education and Experiences

Academic Experience

I am a third year student (6th semester) at National Institute of Technology Durgapur. I have been performing decently in academics from the start and have managed 8.05 CGPA out of 10. Throughout I have opted for an eclectic selection of courses. Few of them being, Introduction to web technologies like to HTML, CSS and JavaScript, and course on software engineering that introduced me to software development workflow and considerations during development cycle.

Courses

Apart from academic courses, I have also enrolled in various online courses to acquire knowledge of real world technologies. And I have listed a few of them here:-

Udacity's Front-End Developer Nanodegree (Free course)

This was one of the most important course that has improved my knowledge of front end development and most importantly core concepts of JavaScript including closure, prototypal based class system in JavaScript, prototype chains etc. This course boosted my involvement in open source.

Coursera Introduction to Graphic Design (Result: 98%)

Though this course was about graphic design. It helped to learn a lot about ideas of typography, visual contrast and compositions. This will certainly help me throughout summer while designing assets for this project.

Courses on image manipulation and vector graphics software

I did courses on Adobe Illustrator and Adobe Photoshop and worked on many projects employing them, one of which included making game art. Here is a link to some of my work -

<https://drive.google.com/folderview?id=0B3GzDUy0ulh-TXE0RTItTnlXN1U&usp=sharing>

Open source Experience

I have been engrossed in open source for a while. In this time, I have contributed to many projects:-

Fluid - Made myself familiar with the infusion source and also preference framework. Commits and PRs -

<https://github.com/fluid-project/infusion/commits?author=PaliwalSparsh>

<https://github.com/fluid-project/infusion/pulls/PaliwalSparsh>

Working on these bugs I learnt jqUnit(FLUID-4137), Grunt and also how to work together with fluid-design and fluid-tech (FLUID-2279 and FLUID-3921).

Processing (p5.js) - I learnt about technologies used to setup a project like npm, grunt-task runner etc which are used by almost all JavaScript projects including infusion framework. We will be using these technologies to setup our project.

<https://github.com/processing/p5.js/commits?author=PaliwalSparsh>

Mozilla and GNOME - These marked as the beginning of my open source journey and helped to learn a lot about people and culture in the open source community.

<https://goo.gl/9VWXvj>

Personal Project

I have worked on many personal projects which can be found on my Github profile -

<https://github.com/PaliwalSparsh?tab=repositories>

Apart from these I developed a mobile game **Pogo Penguin**, the experience I gained during its development will certainly be of great help. I developed the game using Unity game engine and programmed it using Unity script (ECMAScript).

I intend to use Phaser framework in this project, which is also based in JavaScript. I am certain that familiarity with JavaScript based game scripting will help me throughout the summer. I designed most of the game assets using Adobe Illustrator and Photoshop for the project.

<https://play.google.com/store/apps/details?id=com.discoman.pogopenguin&hl=en>

Project Description

Game for first-discovery of preferences aims at developing a game that will help users to discover appropriate digital and learning preferences in an interactive environment and by engaging them in a process of "Learning to learn".

Preferences and the problem

Accessibility to any digital or learning resource heavily depends on the way the resource is delivered to the user. It is required to create an environment that is in accordance with the user's learning preferences and ease of access.

Preferences are choices in accordance to how the user wants to consume this resource. We will classify user's preferences into digital preferences and learning preferences.

DIGITAL PREFERENCES

Problems in accessing digital resources can be caused due to disability, illiteracy, digital illiteracy or aging. We can solve this problem by collecting the user's digital preferences and applying it on all digital interfaces he would come across resulting in a consistent and familiar experience across all digital content he consumes. Some of the digital preferences include –

1. **Content Language** - Language in which content is presented. (This also include sign language through symbols).
2. **Font/Text Settings** - This includes font type, font size, text leading etc.
3. **Theme Settings** - Colour combination user prefers for text and background. User prefers high or low contrast.
4. **Accessibility Settings** - Text to speech (speech rate, speech volume), text for audio ie all sounds and audio information should be presented as text, normal audio volume/rate.
5. **Control Settings** – User prefers mouse or keyboard for interaction. This also includes the specificity for on-screen or physical keyboard.

LEARNING PREFERENCES

Learning preferences are learning and teaching styles that work best for a user. Once we know users natural learning preferences, we can provide them with better services in a familiar environment. Most of the time these preferences align with the digital preferences. But some of them are certainly distinct like Annotations - Does the person require hints while working on a problem. Wherever used preferences would refer to both digital and learning preferences as they are more or less same, but when distinction is required I will define it explicitly.

LEARNING TO LEARN

Learning to learn is methodology where users can themselves try and learn what preferences work best for them. We will establish an environment in our project so that user can learn to learn.

After discussing about digital preferences, learning preferences and learning to learn, it is quite easy to state that knowing user's preferences can be very beneficial. So the next

thing is collecting these preferences. There have been amazing attempts to solve this problem, some of these solutions use Infusion Preference Framework. Preference framework enables us to make tools that can facilitate preference collection from users and apply them to all the interfaces. This improves user experience, resulting in consistent and familiar digital environment which user can navigate without difficulty. There have been multiple implementations of this idea with Preference framework like First Discovery Tool of GPII. Problem is that these implementations do fulfil the objective but lack the interactivity that a software should have with its user. Thus, we need to make this activity of “learning to learn” fun.

According to Wikipedia “A videogame is an electronic game that involves human interaction with a user interface to generate a feedback”. The project will use a videogame to generate feedbacks and those feedbacks will be preferences. Before going through the game implementation we will look over some disabilities that can hinder user experience and how will we overcome these.

ACCESSIBILITY FEATURES FOR USERS WITH DISABILITIES

There are four main type of impairments that can hinder users ability to interact with digital resources -

1. Visual

Example conditions : myopia, blindness.

The game will have a size preference which user can set according to his need. Also we will be employing **ARIA attributes** on all possible HTML elements in the project to help people using screen readers, one of such element is the preference editor it will be discussed in the upcoming sections. The project also consist of text-to-speech as a preference, if selected when a level starts all level related instructions will be narrated to the user. Also the user can hover mouse over sections to hear section specific information and instructions.

2. Motor (ability to operate a controller)

Example conditions : RSI, cerebral palsy, Parkinson’s

The game will provide users with multiple control choices, and the user can choose any of them according to his comfort. Users will be asked for their preference of mouse, keyboard or on-screen keyboard. Also game will not have any reliance on precise movements.

3. Hearing

Example conditions : auditory processing disorder etc.

The game will include subtitles, volume configuration and visual cues. The term video game implies a natural bias towards visual interaction rather than something as hearing

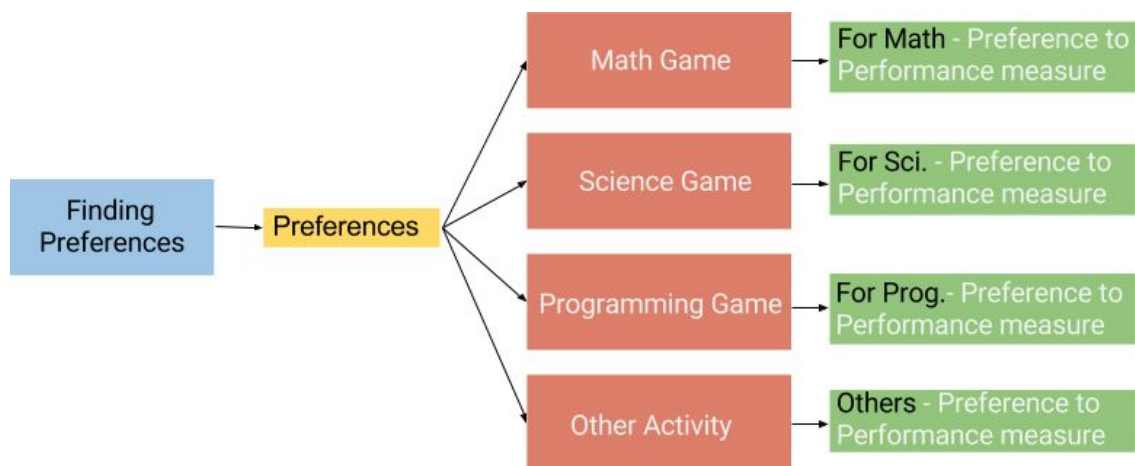
thus the user can always take visual aid to overcome their disability and for this we have tried to employ all the game instruction in text format. So that people with hearing impairment would have a comfortable experience while playing the game.

4. Mobility

Making the game accessible for people who have mobility impairment will require some external hardware. Like for amputees we can establish a system for their voice recognition and response. But including this will not be feasible in limited time during summer. I would love to work on voice recognition as part of continuing work after GSOC.

IMPLEMENTATION

Model of the project is divided into two sections. First section is aimed at preferences collection. Second section measures and tracks user experience with the selected preferences.



SECTION ONE

SECTION TWO

Project will involve working on these two sections. Section one i.e. "Finding Preferences", will consist of six interactive scenes aiming to discover preferences of user. Section two will be a game titled "The Science Game" with five levels that are bond together through a story.

Project will involve working on only one game as it would be infeasible to work on all the games in limited time given as part of Google summer of code. For future development

the model used in game provides a lot of flexibility, it can be extended to different areas of learning like Maths, English etc. by other contributors. Thus users would be able to know about their peculiar preferences for specific learning environment.

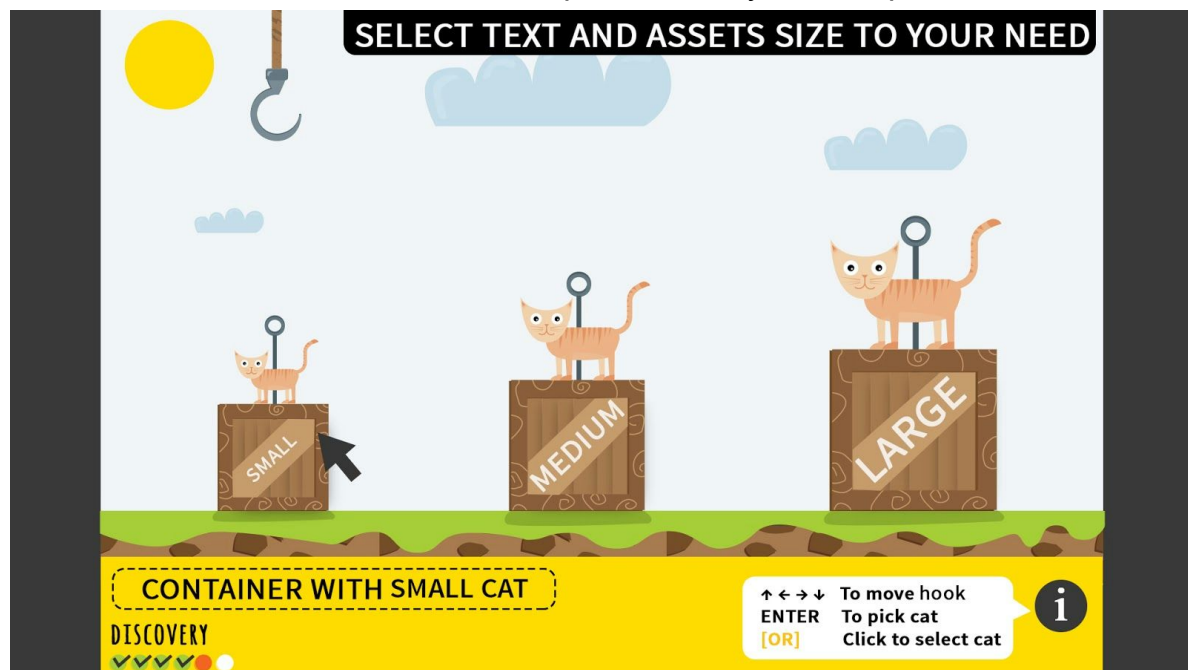
Below mockups are used to explain the implementation of the game. All the artwork in the mockups has been solely designed by me.

SECTION ONE

This section aims at first discovery of preferences, it is divided over six scenes such that each scene employs innovative ways to collect one of the digital or learning preferences as discussed in section on digital preferences and learning preferences. Let's have a look at couple of mockups to have an overview of this section -

This scene will collect text and assets size preference which user finds comfortable. People with myopic vision or other vision related issues would find help not only in identifying text but also different elements in game. The text below describes various features accessible to the user in the scene.

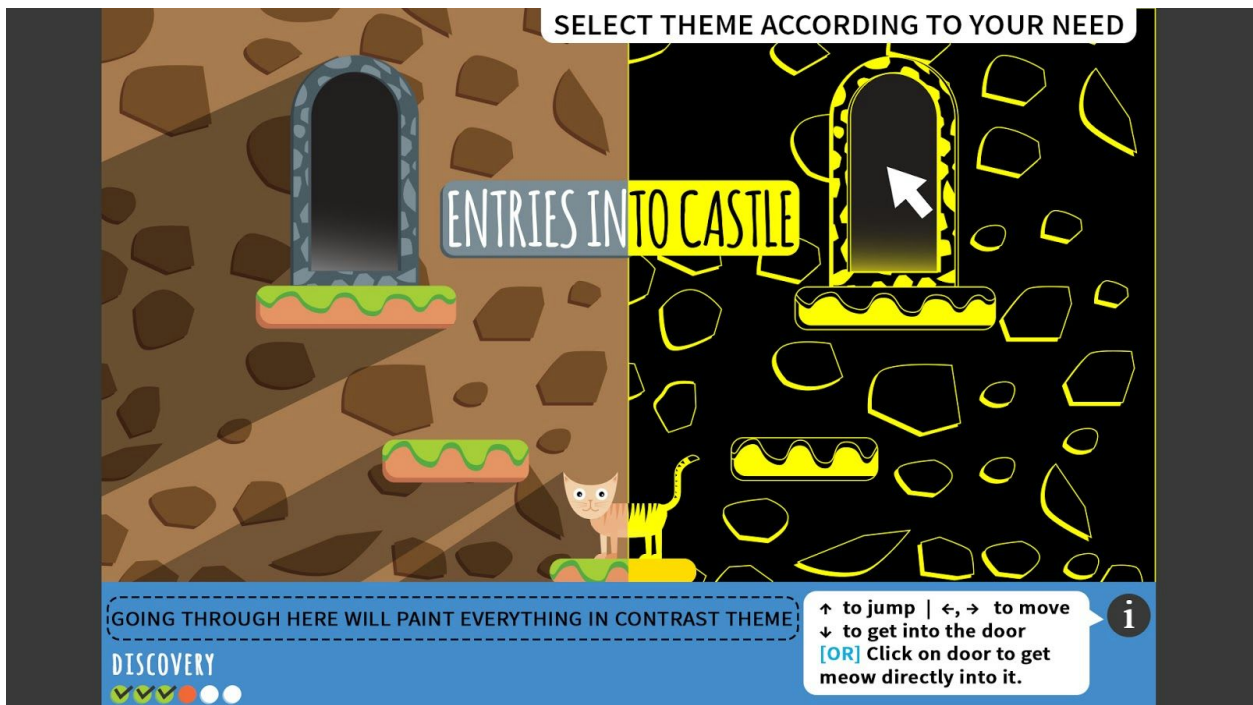
The left bottom of the screen shows that user has already discovered four preferences, and user is right now on the size preference level. Scene description is made clear by the black rectangular box that serves as scene heading on the top right. The text in dotted rectangle box towards bottom tells us about the element on which mouse is hovered. Scene also consists of a tooltip identified by "i" to help the user.



All the preferences that are discovered in previous scenes will be used in current scene. Consider user chose text-to-speech, as soon as the user enters current scene "Select text and assets size to your need" is narrated from the rectangle in the top right. When user hovers over each block "Container with small cat", "Container with big cat" and so on is narrated. This helps people with visual impairments. The text that is narrated on hover is the text in dotted rectangle box in the left bottom.

Also, here tooltip as a help shows controls for keyboard and mouse. Apart from this, scene will also have touch controls that will be identical to that of mouse, like tap on screen will select the cat. So the game provide user with variety of choices like a user can either use keyboard to move hook and grab a box or a user with weak vision can hover and understand the context by listening and simply click on the box. User will also have a option to select their control preferences whether mouse, keyboard, onscreen keyboard or touch.

This another mockup will collect theme preference from user, if he goes through the right door castle will be painted in contrast theme, and if they go through left door castle will be painted in color theme. Basically user will go through the path which he likes and that will be his theme preference. So user will choose the entry to his liking.



SECTION TWO

After section one is completed user will enter the section two that will help user to check discovered preferences against learning environments and improve his choice of preferences accordingly. Section two will achieve this through a game.

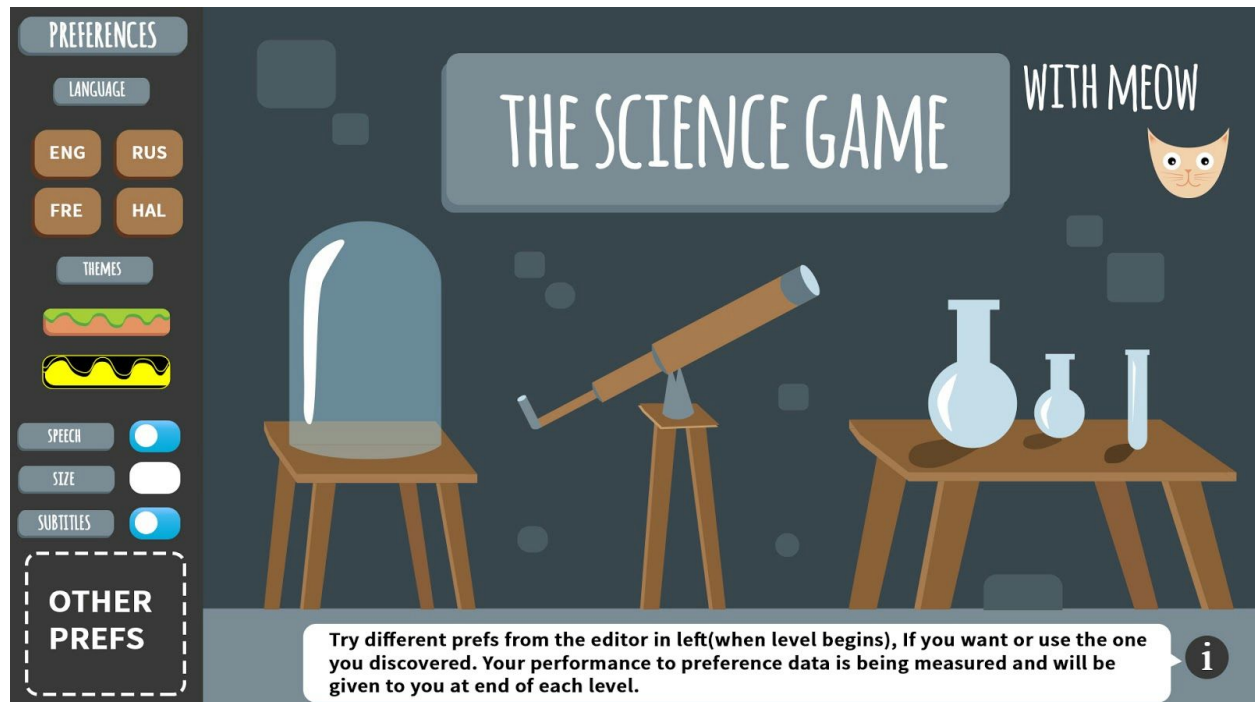
STORY OF GAME

After first section is completed, we will dictate story of Meow the cat, she is the protagonist of our game. There will be a short animation showing game's story which goes as –

“Once upon a time in a happy corner of the world lived a cat named “Meow”. She loved her woollen balls very much but one day a storm came and took away all the wool balls with it. Meow is on her quest to get all her woollen balls.”

After this her quest will start, she will go through “The Science Game” to the destination where she will find one of her woollen ball. During the journey she will need to overcome difficulties like crossing the river, going over the hill and she needs to know about concepts of science to overcome those problems.

Similarly going on other quests will get her all her woollens balls back. But we will be working only on Science game(quest) for now.

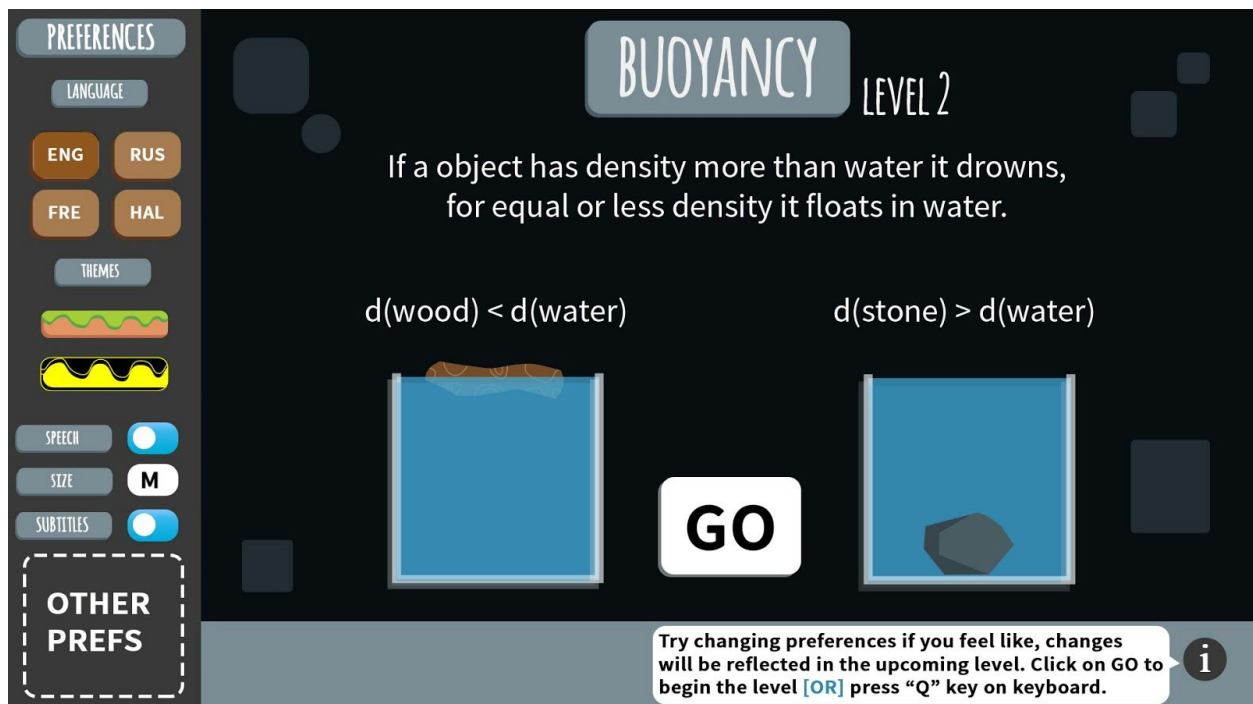


PREFERENCE EDITOR

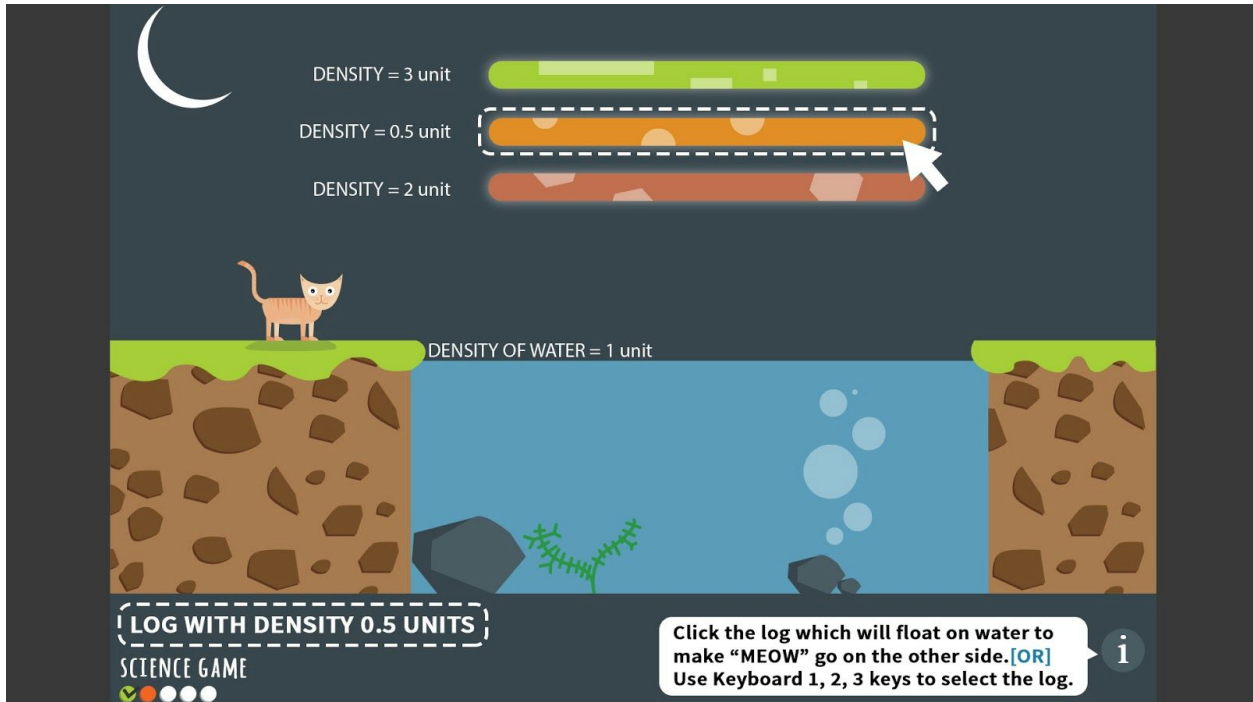
To manage our preferences, we will be using a preference editor. For the six scenes in the Section One preference editor will be completely hidden. Once the user has moved to Section Two - "The Science Game", preference editor will start to appear at specific times, with the help of mockup we will look at when will it appear.

Before every level of the science game, users will be provided with the knowledge required to complete that level, as shown in the mockup. Also scenes providing knowledge to users before each level will be the places in game where preference editor will be visible. Here user can select the preferences according to his need (by default they are the ones that user has discovered in section one) and the preferences selected by user will be reflected in the level, which will start as soon as the user clicks the "GO" button.

Pros of adding the preference editor is that it gives the user a chance to try and understand what preferences are better for him. Mockups below will provide a broad picture of how preference editor is an important part of the strategy to help the user in the process of learning to learn.

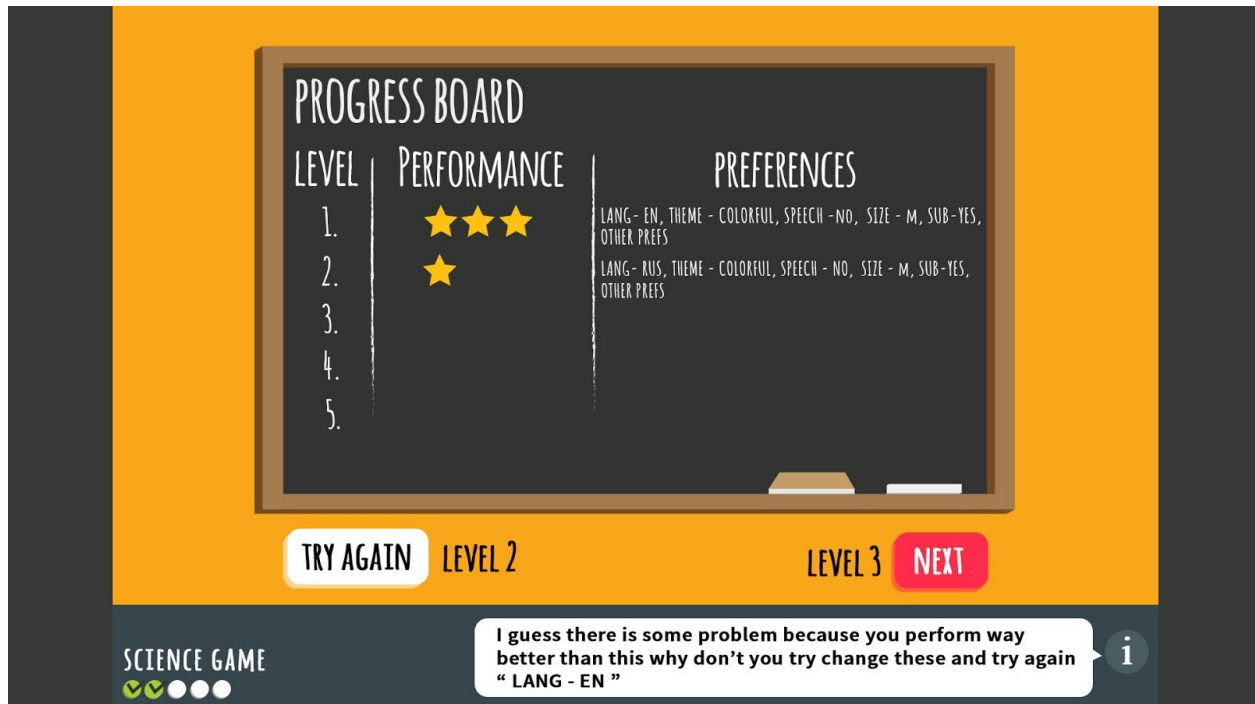


Changes made in the preference editor will be reflected in the following quest level and it can be seen that preference editor is hidden as the level has begun.



In the mockup above answer to the question is one of three options. If the user is not able to answer even after three attempts due to some reason like he is not able to understand the level because of the preferences he is using, we need to ensure the user does not get stuck in the level so we will forward him to the progress board as depicted in the mockup below, where we will suggest him (in the tooltip) with preferences that would work better for him, how we figure out these suggestions we will discuss in the next section. Then he can either choose option to “try again” or go to “next level”. Performance of user will be based on attempts the user takes to solve the quest.

While on Progress board user can evaluate, how they have performed with different choices of preferences and observing these they can easily understand what preference configuration works best for them. By this we are able to provide ability of **LEARNING TO LEARN**. This is made possible with the help of preference editor and progress board.



Imagine we have a user named Tim. Tim thinks he knows two languages English and Russian. During level one Tim had his language preference set as English, he passed the level comfortably, but during level two he changed his language preference to Russian, after going through the level two when Tim was presented with progress board, he realised that when used English he was able to interpret and respond to things easily as compared to when he used Russian, also the game suggests Tim to use "LANG-EN" based on his previous performances. Tim got to learn that English is better for him and we succeeded in our goal to help Tim.

SUGGESTING PREFERENCES

To suggest preferences to users, game will look at all previous performances of the user that means will take a look at levels where he performed well and collect the common preferences in all those levels. This increases the probability that these preferences work for the user. Preference choices that are deteriorating users performance can be found by look at levels where he did not perform as well. Suppose the user does not perform well in the very first level. This means that the preferences chosen by the user during the Preference Discovery are not ideal for him. For the next level the game will suggest him with preferences with slight alterations to his current preferences.

DEVELOPMENT

PHASER

Phaser will be used as HTML5 game framework for the project. Phaser is open source and has great community support, thus stabilizes itself to be an easy choice. Phaser has support for things like Phaser.touch library which will help us implement touch into the game.

PREFERENCE FRAMEWORK

We will be using the Infusion preference framework to make the Preference Editor, we will be making our custom prefEditor in preference framework, add required preferences to it accordingly, and manage CSS such that it aligns with the game design.

PHASER AND PREFERENCE FRAMEWORK

We have discussed about the preference editor and phaser next point of discussion is how to use them together. We would be providing a wrapper/binding for Phaser so that we can use Phaser in accordance to Infusion practices. Also we will setup a model for the game to store all its states and preferences, so that we can access this model externally, this is another important infusion practice. We will be looking over all these implementation details in depth during the community bonding period.

TESTING

Testing helps to point out defect and errors during the development phase, ensuring that as we are going ahead we don't break things that are already implemented. Thus there is a need to implement testing in the project's workflow.

For unit testing we will be using infusion's jqUnit testing framework, I have developed familiarity with it while working on FLUID-4137. Testing will be divided in two parts writing tests for the game and tests for the preference framework. I have considered both in the timeline.

Integration Testing using IoC testing framework would serve as a bonus track for the project. If time permits, I will be working on it. If not possible during the summer, I will be working on it after GSOC.

Timeline

Community Bonding Period (22 April - 22 May)

I will spend most of my time to learn Infusion and the Preference Framework. Apart from this I will iterate over my design mockups for the project, and will try to perfect game design. Also I will discuss implementation of preference framework (implementing saving preferences, server integration and other things as required) with mentors.

Week 1 (23 May - 29 May)

- Setting up the project (npm, git, jqUnit, grunt).
- Design assets for level 1 - 3 .
- Make level 1 in phaser and write tests for it.

Week 2 (30 May - 5 June)

- Make preference editor and attach to preference in level 1(will do same for all levels).
- Write tests for preference editor.

Week 3 (6 June - 12 June)

- Make level 2 - 4 in phaser and write tests.
- Design assets for level 4 - 6.

Week 4 (13 June - 19 June)

- Make level 5 - 6 in phaser and write tests.
- Fix bugs.

Week 5 (20 June - 26 June)

- Buffer week for mid term evaluation.

MID TERM EVALUATION:

Deliverable - Section One of the game along with the preference editor.

Week 6 - Week 7 (27 June - 10 July)

- Start working on "The Science Game".
- Design assets for level 1 - 2 (Science game).
- Work on preference editor styling to match game design.
- Code level 1 - 2 in phaser, test them, and make preference editor work with it.

Week 8 - Week 9 (11 July - 24 July)

- Work on Progress Board including design, code and tests.
- Design assets for level 3 - 4.
- Code level 3 - 4 in phaser.

Week 10 (25 July - 31 July)

- Buffer week to come over any delay in work.

Week 11 - Week 12 (1 August - 14 August)

- Work on assets for level 5.
- Code level 5 in phaser and test them.
- Documenting the project properly.
- Work on bugs.

Week 13 (15 August - 21 August)

FINAL WEEK

- End time fixes, improve documentation.
- Submit code sample.

Bonus Track : Including Integration tests with IoC framework if time permits.

Deliverables/Outcomes

Upon completion the project will deliver a game that-

1. Helps you with your first discovery of preferences, in a fun manner.
2. Enables you to keep a check on your performance while working with those preferences.
3. Lets you change your preferences while playing the game to find the preferences that work best for you.
4. Maintains a Progress Board to keep a record of your progress and performance for the different preference combination you have tried throughout the game.
5. Implements suggestion system to suggest you better preferences for yourself.

Questions

How will you document your weekly progress?

Weekly progress of the project will be document on my blog www.paliwalsparsh.wordpress.com. Blog will include the following section -

1. Work for the previous week and how we implemented it.
2. Interesting issues that we faced and how we resolved them.
3. New technologies or any other thing we learned.

How will you ensure your project code continues to be maintained, after GSoC?

The model used to implement Section Two of the project is robust, and it will be easy to integrate new games into the project. Thus, it will be easier for new contributors to come and add their own games to the project. After completion of GSoC phase, I will contribute the project to fluid-project's github space. I will ensure that I am there to maintain the project and help new contributors. I will try to involve some of my college mates into the project as well, upon completion. Thus making them familiar with open source, new technologies and the community.

What is your Github account?

<https://github.com/PaliwalSparsh>

Will you be working on your own, or will you have partners?

I will be working on the project alone. The expert guidance from the mentors would be enough for me to work on this project, working with them has already been fun for small contributions that I have made to fluid-project.